Capella to TASTE MBSE bridge

Project Team:

Presenter: Michał Kurowski
N7 Space Sp. z o.o.
mkurowski@n7space.com

Other project contributors: Arkadiusz Wójcik
was with *Creotech Instruments S.A.*
arkadiusz.wojcik@creotech.pl

Michał Kocon
N7 Space Sp. z o.o.
michal.kocon@n7space.com

Daniel Kaczmara
Creotech Instruments S.A.
daniel.kaczmara@creotech.pl

Bartłomiej Juszczyk
Creotech Instruments S.A.
bartłomiej.juszczyk@creotech.pl

Michał Mosdorf
N7 Space Sp. z o.o.
mmosdorf@n7space.com

ESA team: Hans Peter de Koning
ESA (retired)

Maxime Perrotin ESA
Maxime.Perrotin@esa.int

Short Abstract: The presented project provides a bridge between Capella, an open-source MBSE tool supporting the Arcadia method, and TASTE, ESA's open-source MBSE toolchain. The bridge is implemented via a Capella plugin, which translates the Capella's data and physical architecture models into TASTE-compatible ASN.1 and AADL models, which can be further enhanced with behaviour definitions through C, Ada or SDL, and compiled into deployable binaries. The bridge was validated by implementing software for a Mass-and-Thermal Mockup running on STM32 MCU.

Keywords: Capella, TASTE, Arcadia, ASN.1, AADL, SDL, MSC, MBSE, STM32, plugin

Background. Capella [1], originally implemented by Thales, is an Eclipse-based tool implementing the Arcadia method [2], allowing to perform operational need analysis, system analysis, logical architecture design, physical architecture design and finally define a product breakdown structure, providing an alternative to UML and SysML. It allows to capture requirements and other project specific metadata, delivering a high-level cross-domain MBSE solution. While it does not provide any code generation capabilities by itself, it is highly extensible through Java plugins. TASTE [3], managed by ESA, is a set of tools focused on supporting model-based software development. In particular, it allows to generate code from ASN.1, AADL and SDL models, which can be then compiled and deployed onto the supported platforms, including x86, Leon3 and ARM STM32. The resulting software can be then tested using executable (via Python) MSC diagrams. A bridge connecting the two solutions, in the form of a Capella plugin, was implemented during the *MBSE_Implement* project founded by the European Space Agency and carried out by Creotech Instruments (prime contractor) and N7 Space (subcontractor). It allows to apply an MBSE based approach throughout the entire software product lifecycle, from high-level cross-domain analysis to implementation, testing and deployment.

Bridge implementation. After discussions held between Creotech Instruments, N7 Space and ESA, N7 Space analysed the scope and explicitness of Capella model elements with respect to the capabilities and requirements of TASTE toolchain. The following was considered – data model, architecture and behaviour.

Capella's data model focuses on the data semantics. TASTE on the other hand models both the semantics, via ASN.1, and encoding, down to the bit-level, via additional ACN definitions or by application of default UPER rules. While Capella's data model could be enhanced with additional metadata for bit-level encoding specification, therefore enabling ACN generation, it was considered complicated and unnecessary. N7 Space implemented ASN.1 generation, by mapping Capella's Packages, Classes, Unions, Collections, Numeric Types into ASN.1 Modules, Sequences, Choices, Sequences Of and Integers or Reals respectively. The user can additionally choose, through custom properties, an encoding specification from between UPER, platform native or ACN. In case of UPER and platform native, encoding is handled automatically by TASTE. In case of ACN, the additional rules must be provided by the user separately. Capella's data model supports data types, values and expressions. On the other hand, ASN.1 supports only data types and values. In order to partially resolve this limitation, a simple evaluator was implemented in the plugin to translate integer expressions into concrete values. As Capella's Units, relevant to Physical Quantities, do not have a

corresponding construct in ASN.1, they were implemented via ASN.1 type name postfixes. Similarly, Capella's class inheritance hierarchy is translated into a set of ASN.1 Choices.

Capella's architecture model is expressed via logical and physical architectures. The former is usually considered a "principle", coarse-grained, general architecture. The latter is the finalized architecture. As TASTE requires a concrete architecture definition, and the tracing between the physical architecture and the logical architecture is maintained in Capella anyway, the physical architecture was chosen as the base for AADL generation. N7 Space implemented a mapping from Capella's physical Nodes, Components/Actors, Links/Paths, Ports, Functions and Functional Exchanges into TASTE Nodes, Partitions, Buses, Devices, Functions and Interfaces respectively. The TASTE concepts are expressed via standard AADL constructs such as Packages, Systems, Processes, Subcomponents, Connections, Features and Subprograms. As time-and-space partitioning is not supported in the plugin, all components residing on a single node are merged into a single partition. An extensible and explicit mapping to target processors and drivers is provided through custom string properties. The developed mapping allows the generation of TASTE Interface and Deployment Views.

While Capella supports the modelling of behaviours through Sequence, as well as Mode and State diagrams, N7 Space deemed their translation into SDL (or any other executable language) infeasible without significantly extending the Capella's model. Consequently, the concrete behaviour definition is to be performed directly in TASTE, e.g. via SDL, C or Ada. Unambiguous, formal and user-friendly software behaviour definition within Capella can be a subject for future work.

The plugin implemented by N7 Space first checks the Capella model for consistency and completeness from TASTE's perspective (thus constraining the Capella's model to a subset with well-defined semantics), provides feedback, allows the user to select data or architecture models' subsets and then generates the corresponding ASN.1 and AADL artefacts. The plugin allows to naturally follow the Arcadia method (implemented in Capella) with TASTE based implementation within a fully model based workflow. A partial approach is also possible by using only the generated ASN.1 [4].

Validation. In order to validate the plugin and the MBSE approach, a use case scenario was established jointly by Creotech and N7 Space. Creotech modelled a Mass-and-Thermal Mockup software in Capella. The model was then iteratively refined using feedback from N7 Space, illustrating the benefits of model-based design formalization and disambiguation for achieving common understanding across different industrial partners. After the finalization, the model was automatically translated into ASN.1 and AADL files using the plugin. N7 Space implemented the software behaviour in SDL (high-level functionality), C (peripheral drivers) and Ada (RS-485 communication device driver). The C code was based on an alternative, twin software, manually coded by Creotech Instruments. The software was then successfully deployed and tested on a physical Mass-and-Thermal Mockup designed and produced by Creotech for an in-house developed satellite platform. The test scenarios, defined by Creotech, were implemented by N7 Space in Python using the code automatically generated from MSC diagrams created in TASTE. Additionally, as a part of the applied MBSE approach, Creotech Instruments used a freely available M2Doc [5] add-on to generate documentation from an internally developed Capella model.

Summary. The developed plugin allows to translate a well-defined subset of Capella's data and physical architecture models into ASN.1 and AADL models compatible with TASTE. These models can be then further enhanced with behaviour definition via SDL, C or Ada, and compiled into executable software for target platforms. The plugin and the MBSE approach were successfully validated via the implementation of software for a Mass-and-Thermal Mockup. Feedback from the validation was propagated as suggestions or remarks to the TASTE project.

References:

[1] Capella, https://www.eclipse.org/capella/

[2] Model-based System and Architecture Engineering with the Arcadia Method, Jean-Luc Voirin, ISBN 978-1-78548-169-7

[3] TASTE, https://taste.tools/

[4] Maxime Perrotin et al.. TASTE in action. 8th European Congress on Embedded Real Time Software and Systems (ERTS 2016)

[5] M2Doc Capella add-on https://www.m2doc.org/